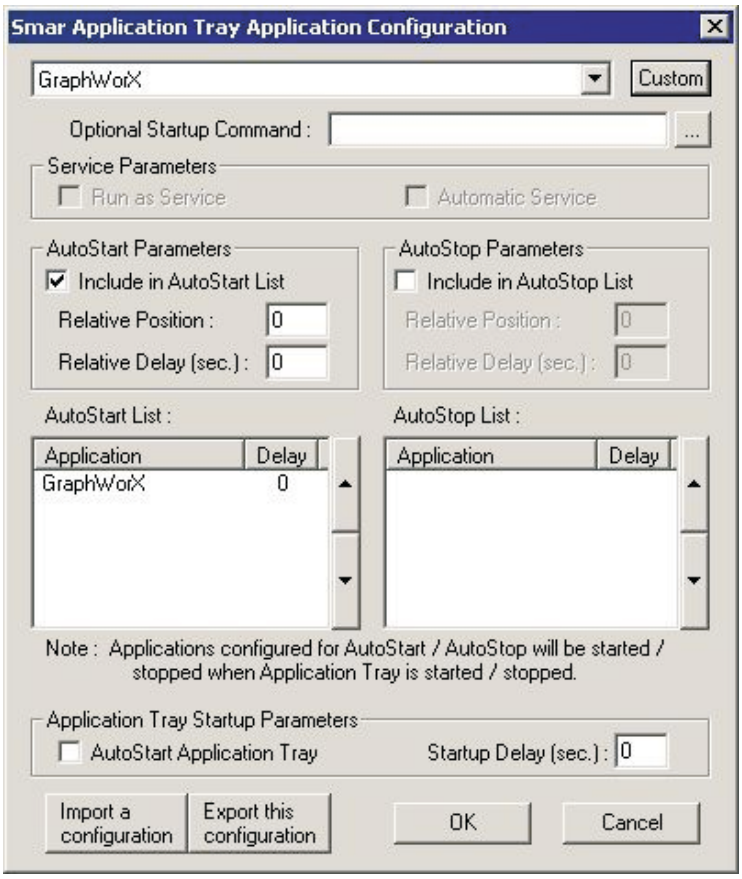
- ProcessView Tray

Smar
First in Fieldbus

ProcessView Tray



MAY / 06
ProcessView Tray
VERSION 8

FOUNDATION

P V I E W P T Y M E

www.smar.com

Specifications and information are subject to change without notice.
Up-to-date address information is available on our website.

web: www.smar.com/contactus.asp

TABLE OF CONTENTS

INTRODUCTION TO PROCESSVIEW TRAY..... 5
STARTING PROCESSVIEW TRAY.....5
STARTING AND STOPPING APPLICATIONS.....6
STARTING APPLICATIONS IN PROCESSVIEW TRAY 6
STOPPING APPLICATIONS IN PROCESSVIEW TRAY 6
VIEWING APPLICATION STATUS 7
AUTO START AND AUTO STOP 8
CONFIGURING AUTO START AND AUTO STOP PARAMETERS 9
CONFIGURING APPLICATION SETTINGS.....11
CUSTOMIZING THE APPLICATION ITEMS LIST 12
IMPORTING AND EXPORTING CONFIGURATIONS 16
RUNNING APPLICATIONS AS NT SERVICES 17
OLE AUTOMATION 18
PROPERTIES 20
USING PROCESSVIEW TRAY IN PROJECTWORX.....20
STARTING APPLICATIONS IN PROJECTWORX..... 21
STOPPING APPLICATIONS IN PROJECTWORX 22
SETTING APPLICATION PROPERTIES IN PROJECTWORX..... 22
VIEWING APPLICATION PARAMETERS IN PROJECTWORX 24

INTRODUCTION TO PROCESSVIEW TRAY

The ProcessView Tray Utility provides convenient a way to either manually or automatically start and/or stop various ProcessView applications, including the following:

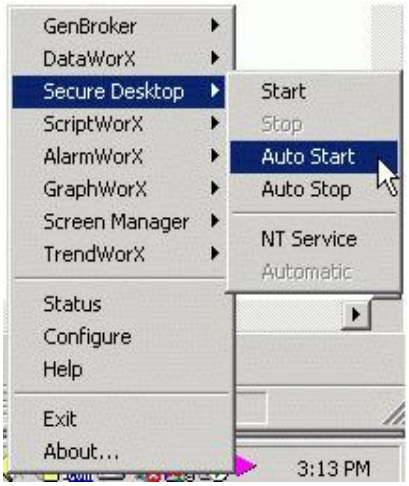
- GraphWorX
- TrendWorX
- AlarmWorX
- GenBroker
- Secure Desktop
- DataWorX
- Screen Manager
- ScriptWorX

ProcessView Tray is a simple user interface located in your Windows taskbar, making it easy to access at any time. The ProcessView Tray executable file (**Gentray.exe**) should be placed in your computer's **Startup** group so that it is available every time your computer is started.

Starting ProcessView Tray

To launch ProcessView Tray:

1. From the Windows **Start** menu, select **Programs** > **Smar ProcessView > Tools > ProcessView Tray.**
2. ProcessView Tray is launched, and a purple triangle icon appears in the in the lower right corner of the Windows task bar, as shown in the figure below. Click on the triangle to open the ProcessView Tray user interface.



ProcessView Tray User Interface

The user interface consists of a pop-up menu that lists various applications in ProcessView. Selecting one of these items shows any application items in a pop-up menu. For each item, there is a list of commands to **Start** or **Stop** and **Auto Start** or **Auto Stop** the application. Some applications have the option of running as Windows NT service.

Other commands on the menu include the following:

- Status: Displays a list of ProcessView applications that are currently in runtime mode.
- Configure: Launches the ProcessView Tray Application Configuration dialog box.
- Help: Launches the online Help documentation.
- Exit: Closes the ProcessView Tray application.
- About: Launches the About Box, which contains information about the product version number, copyright, and available disk space. It also contains information about how to contact the company.

Starting and Stopping Applications

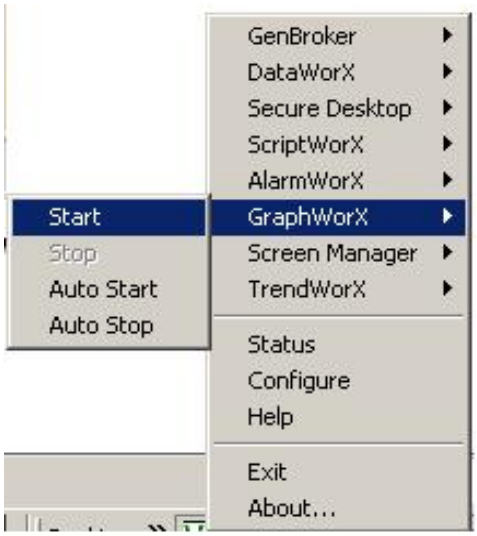
ProcessView Tray provides a way to either manually or automatically start and/or stop applications within ProcessView. In the ProcessView Tray user interface, you can specify which applications to start and stop. You can also set up applications to auto start upon Windows startup, setting their timing and launching position parameters.

NOTE
ProcessView Tray does not detect applications that are open but are not in runtime mode. An application can be detected as running only if the application is launched through ProcessView Tray.

Starting Applications in ProcessView Tray

To launch an application from the ProcessView Tray user interface:

1. Select the application and choose **Start** from the pop-up menu, as shown in the figure below.



Starting an Application in ProcessView Tray

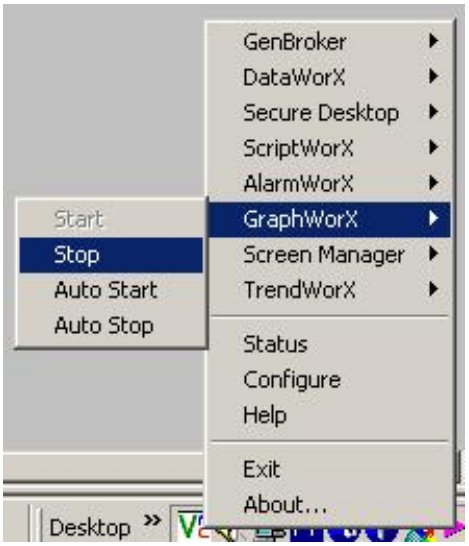
2. The application is launched, and its status changes to Running, as shown in the figure below.

Application Running in ProcessView Tray

Stopping Applications in ProcessView Tray

To shut down a running application from ProcessView Tray:

1. Select the application and choose **Stop** from the pop-up menu, as shown in the figure below.



Stopping an Application in ProcessView Tray

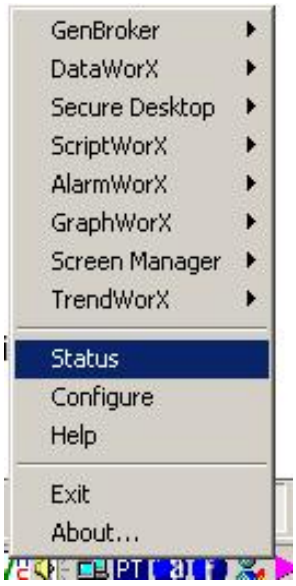
2. The application is shut down, and its status changes to **Not Running.**

Viewing Application Status

When you start or auto start an application from ProcessView Tray, the application opens in runtime mode. An application can be detected as running only if the application is launched through ProcessView Tray. ProcessView Tray does not detect applications that are open but are not in runtime mode.

To view the status of applications in ProcessView Tray:

1. Select **Status** on the ProcessView Tray user interface, as shown in the figure below.



Viewing Application Status

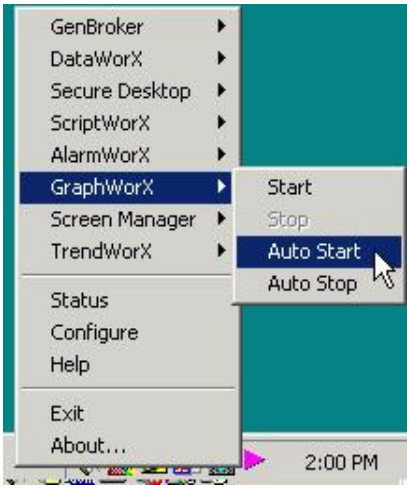
2. The **Status** box opens, as shown in the figure below, listing any open applications that are currently in runtime. If no applications are currently running, the status box says "no applications running."

Application Running in ProcessView Tray

Auto Start and Auto Stop

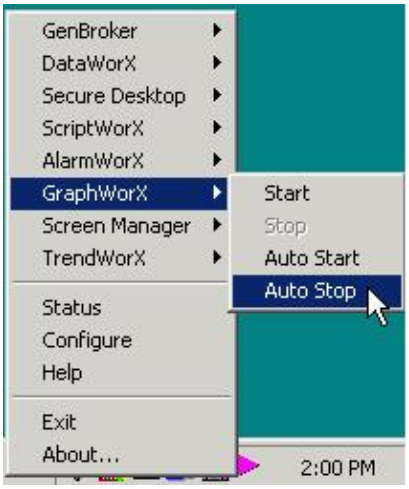
The **Auto Start** and **Auto Stop** options in ProcessView Tray enable you to automatically open and shut down ProcessView system applications in an orderly fashion.

When **Auto Start** is selected for an application in the ProcessView Tray user interface, as shown in the figure below, the application is automatically launched upon ProcessView Tray startup.



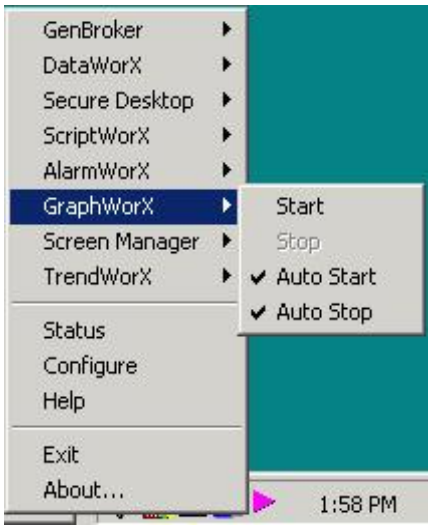
Auto Starting an Application in ProcessView Tray

When **Auto Stop** is selected for an application in the ProcessView Tray user interface, as shown in the figure below, the application is automatically closed upon ProcessView Tray shutdown.



Auto Stopping an Application in ProcessView Tray

Once **Auto Start** or **Auto Stop** has been enabled for an application, a check mark appears next to each menu item, as shown in the figure below.

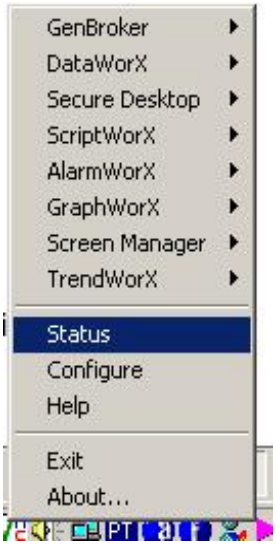


Auto Start/Stop Enabled for Application

NOTE
For more information about auto start/stop settings, please see "Configuring Auto Start and Auto Stop Parameters".

Configuring Auto Start and Auto Stop Parameters

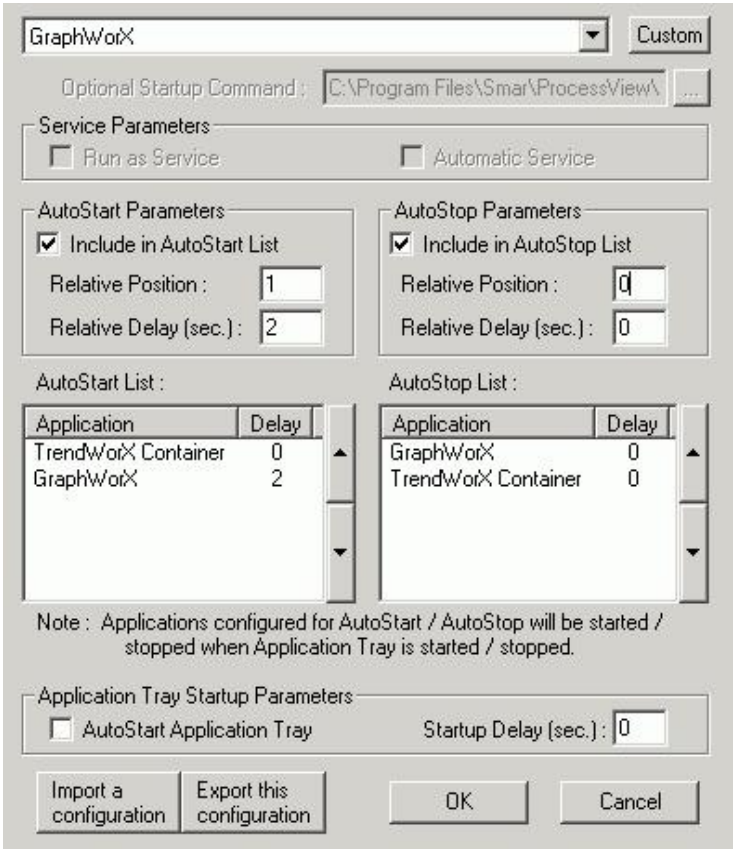
When auto start or auto stop is enabled for multiple applications, ProcessView Tray creates a management list for applications that are auto-started or auto-stopped. To configure the auto start/auto stop parameters, select **Configure** on the ProcessView Tray user interface, as shown in the figure below.



Viewing the ProcessView Tray Configuration Settings

This opens the **ProcessView Tray Application Configuration** dialog box, as shown in the figure below. Any items already selected for Auto Start or Auto Stop will already appear in the **Auto Start List** or the **Auto Stop List,** respectively. To add an application to these lists, select the application from the drop-down list at the top-left corner of the dialog box and check the **Include in Auto Start List** check box under **Auto Start Parameters** and/or the **Include in Auto Stop List** check box under **Auto Stop Parameters.**

NOTE
When an application is run as an automatic NT service, the auto start and auto stop parameters for that application are disabled.



Configuring Auto Start and Auto Stop Parameters

The order in which ProcessView Tray will auto start or auto-stop applications is determined by their relative start and stop positions specified in the parameters. The **Relative Position** settings begin with 0. Thus, an application with a relative start position of "0" will be the first application to auto start. An application with a relative start position of "1" will be the second application to auto start, and so on. Under the **Auto Start List** and **Auto Stop List,** use the **Up/Down** arrow buttons to change the sequence in which ProcessView Tray will launch the applications.

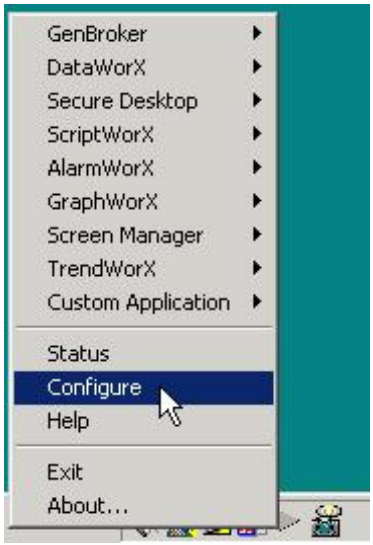
The timing by which ProcessView Tray will auto start or auto-stop applications is determined by the **Relative Delay** settings. You can specify a delay time (in seconds) for each application.

For example, the ProcessView Tray configuration settings for GraphWorX are shown in the figure above. In the **Auto Start List,** GraphWorX is listed second, meaning that it will be the second application to be auto started when ProcessView Tray is launched. The **Relative Position** for GraphWorX is "1." The **Relative Delay** for GraphWorX is 2 seconds, meaning that GraphWorX will be auto started 2 seconds after ProcessView Tray is launched, and 1 second after the Alarm Server (AWXSvr32, which is first in the Auto Start List and has a relative delay of 1 second) will be auto started.

NOTE
When the Auto Start ProcessView Tray check box is checked under the **ProcessView Tray Startup Parameters,** the ProcessView Tray application (GenTray.exe) will automatically launch upon system startup. You can also specify a **Startup Delay** (in seconds) for ProcessView Tray startup.

Configuring Application Settings

To change the configuration settings for ProcessView Tray, select **Configure** on the ProcessView Tray user interface, as shown in the figure below.

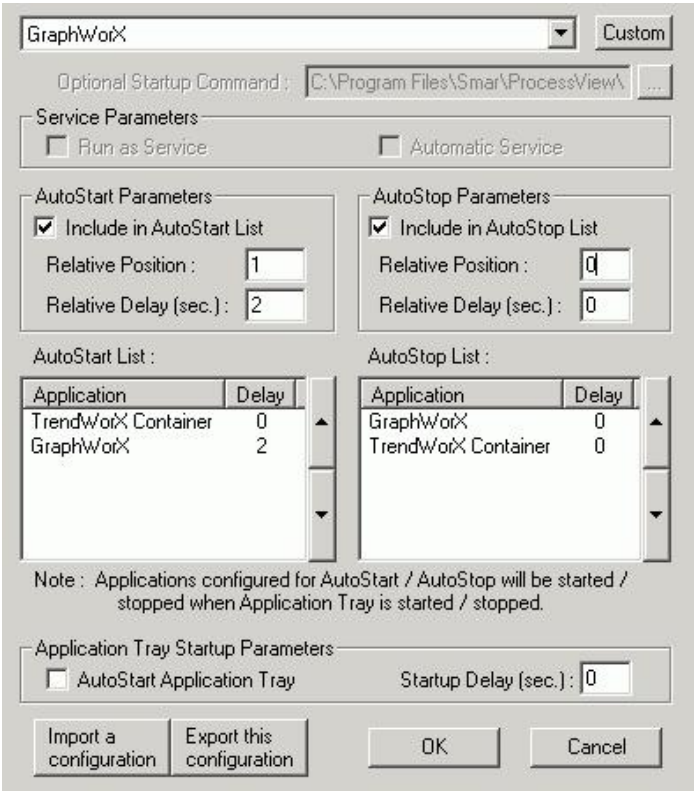


Viewing the ProcessView Tray Configuration Settings

This opens the **ProcessView Tray Application Configuration** dialog box, as shown in the figure below. To configure settings for a specific application, select the application from the drop-down list at the top-left corner of the dialog box (GraphWorX is selected in the figure below).

In the **Optional Startup Command** field, you can specify a startup file for the selected application. For example, you may want to start GraphWorX with a specific display (.gdf) file in runtime mode. To select a startup file, click the **...** button to the right and browse for the file. Select a file and then click **OK.** The directory path and file name appear in the **Optional Startup Command** field, as shown in the figure below.

Click the **Custom** button to customize the application items drop-down list. For more information, please see "Customizing the Application Items List."



Configuring Application Settings

In the **Application Configuration** dialog box, you can also set the NT service parameters for those applications that can run as an NT service by checking the **Run As Service** check box. Please see the "Running Applications as NT Services" section for more information.
Any items already selected for Auto Start or Auto Stop will already appear in the **Auto Start List** or the **Auto Stop List,** respectively. To add an application to these lists, select the application from the drop-down list at the top-left corner of the dialog box and check the **Include in Auto Start List** check box under **Auto Start Parameters** and/or the **Include in Auto Stop List** check box under **Auto Stop Parameters.** For more information, please see "Configuring Auto Start and Auto Stop Parameters."

NOTE
When an application is run as an automatic NT service, the auto start and auto stop parameters for that application are disabled.

If you wish to save the current ProcessView Tray configuration as a separate file, click the **Export This Configuration** button on the **Application Configuration** dialog box. For more information, please see "Importing and Exporting Configurations."

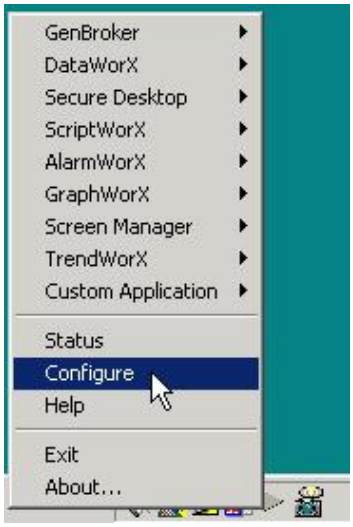
Click **OK** on the **Application Configuration** dialog box to saves the active configuration file in the Windows registry. Each time ProcessView Tray is opened, the most recently saved configuration will be used.

IMPORTANT NOTE
If you want the configuration to be updated in the registry, you *must* click **OK.** If you do not click **OK,** ProcessView Tray will revert to the previously saved configuration the next time it is opened.

Customizing the Application Items List

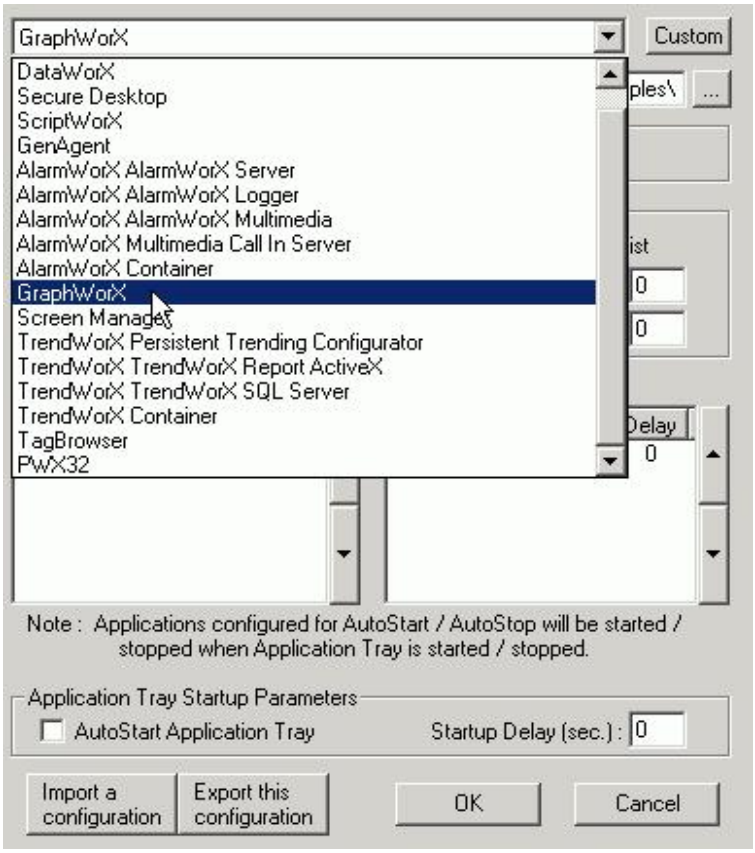
The **ProcessView Tray Application Configuration** dialog box allows you to choose from a list of applications to configure. You can customize this applications list by doing the following:

1. Select **Configure** on the ProcessView Tray user interface, as shown in the figure below.



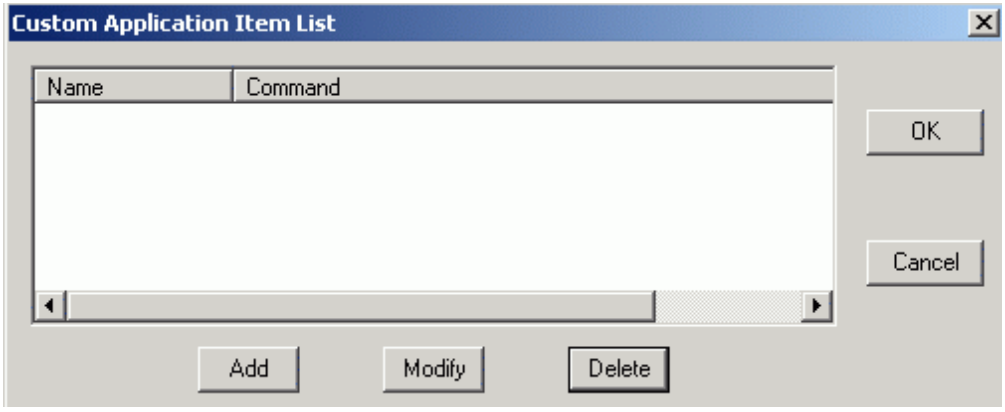
Viewing the ProcessView Tray Configuration Settings

2. This opens the **ProcessView Tray Application Configuration** dialog box. The application items drop-down list at the top-left corner of the dialog box, shown in the figure below, is used to select which application to configure. To modify this list, click the **Custom** button.



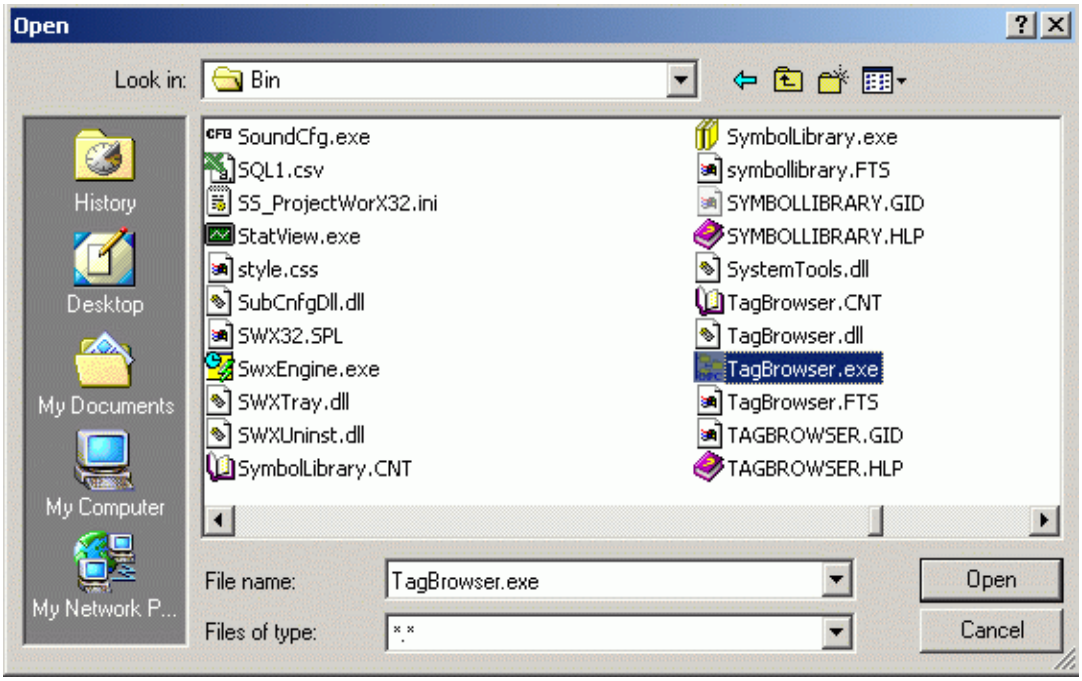
Customizing the Application Items List

3. This opens the **Custom Application Item List** dialog box, as shown in the figure below. To add an item to the list, click the **Add** button.



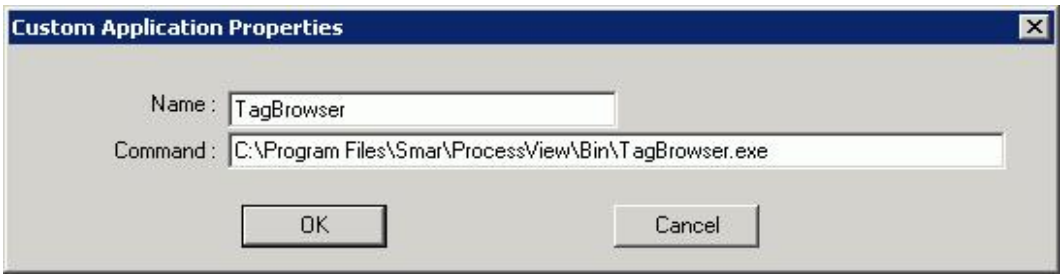
Custom Application Items List

4. Browse for a file to add to the list, select the file, and then click **Open,** as shown in the figure below. In this example, the OPC Tag Browser is selected.



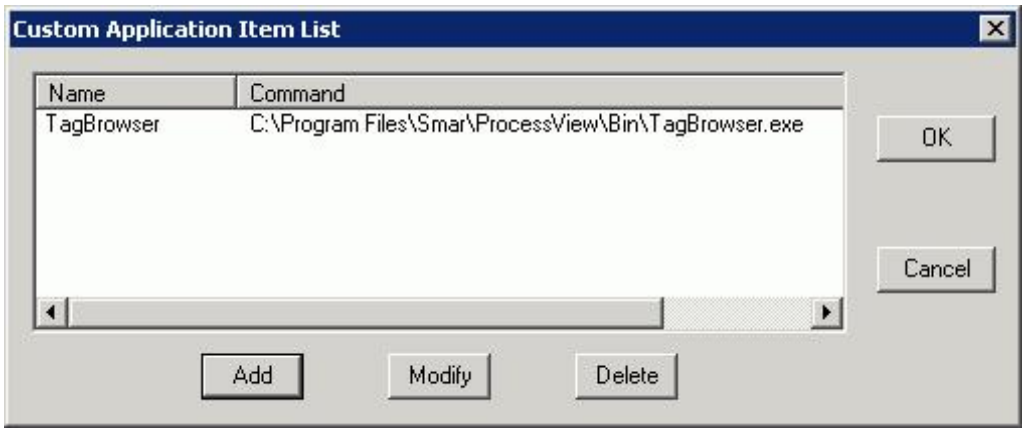
Choosing a Custom Application To Add

5. The **Custom Application Properties** dialog box appears, as shown in the figure below, displaying the selected file and directory path in the **Command** field. In the **Name** field, type the name for the custom application that will be displayed in the ProcessView Tray user interface. Click the **OK** button.



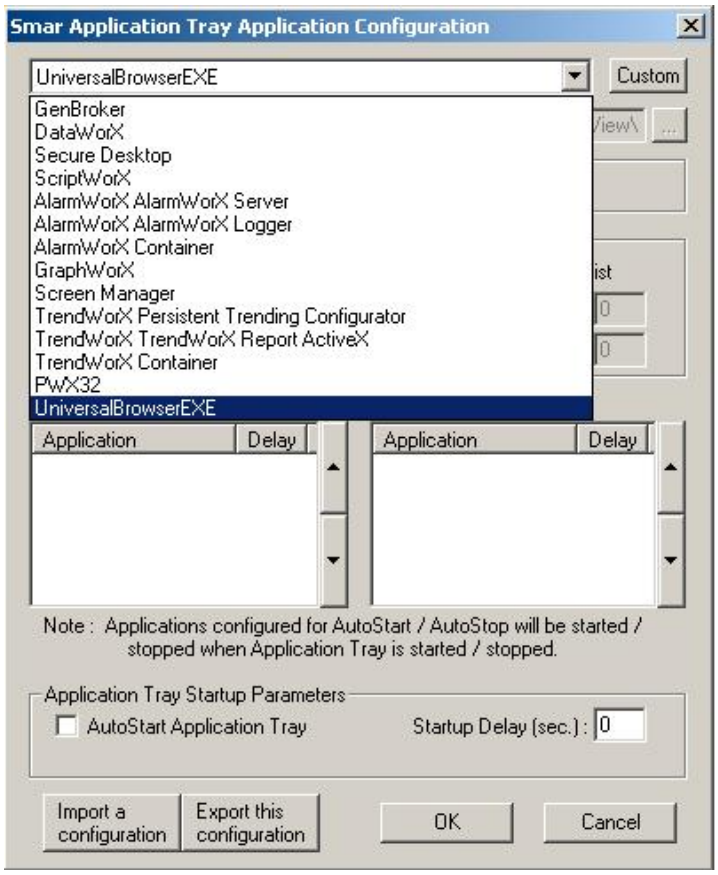
Custom Application Properties

6. The new application is added to the list, as shown in the figure below. At this point, you can also **Modify** or **Delete** the item, or you can add more items to the list. Click **OK.**



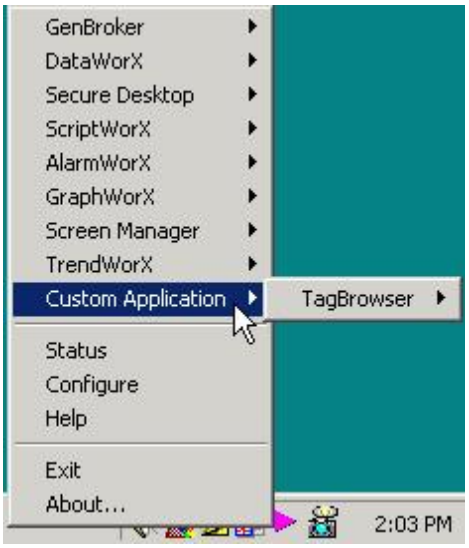
Item Added to Custom Application Items List

7. The **ProcessView Tray Application Configuration dialog** box reappears. The reference ID of your newly added application item now appears in the drop-down list at the top-left corner of the dialog box, as shown in the figure below. You can select the new item from the list and configure its settings. Click **OK.**



New Application Item Added to Drop-Down List

8. The next time you open the ProcessView Tray user interface, a new **Custom Application** option appears on the menu, as shown in the figure below. Selecting **Custom Application** opens a pop-up menu option customized applications, allowing you to start, stop, auto start, or auto stop your application.

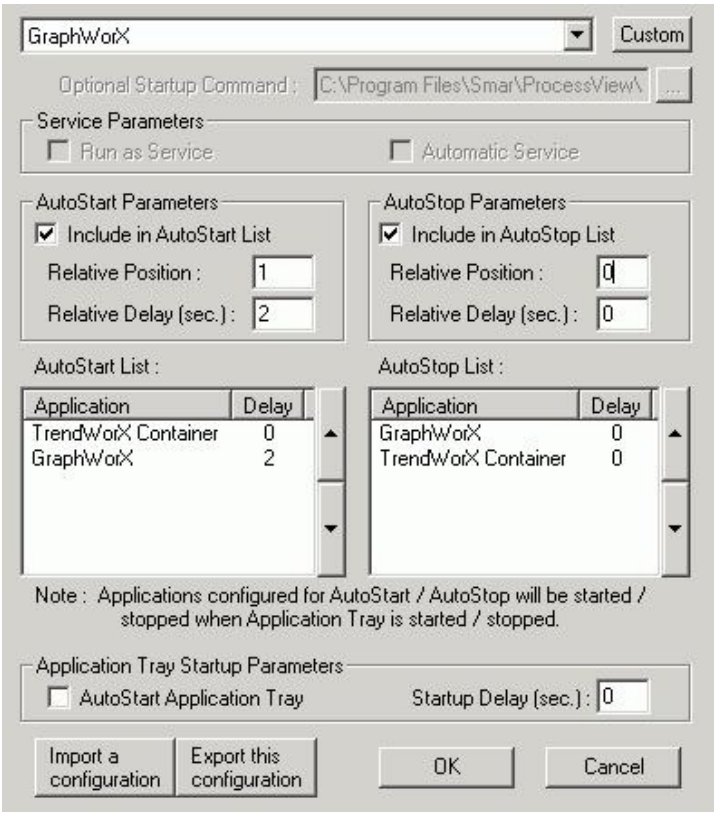


Custom Application Menu in ProcessView Tray Interface

Importing and Exporting Configurations

If you wish to save the current ProcessView Tray configuration as a separate file, click **Export This Configuration** on the **Application Configuration** dialog box, as shown in the figure below. This opens the **Save As** dialog box, which enables you to save a ProcessView Tray configuration (.gtr) file. This way you can archive the configuration for later use on the same PC, for use on a different PC, or for different systems and users, for example.

Once you have archived a configuration as a .gtr file, you can recover the configuration by clicking **Import a Configuration.** This opens a dialog box that allows you to browse for the .gtr file.



Configuring Application Settings

Running Applications as NT Services

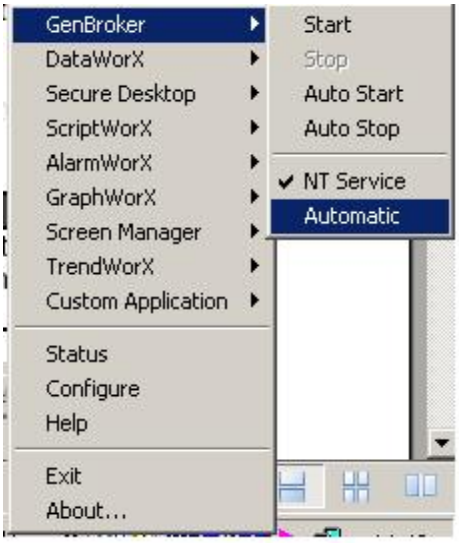
ProcessView Tray also enables you register and run some ProcessView applications as a Windows NT service.

To convert an application that can be run as a service into an NT service:

1. Select the application and choose **NT Service** from the pop-up menu, as shown in the figure below.

Registering an NT Service

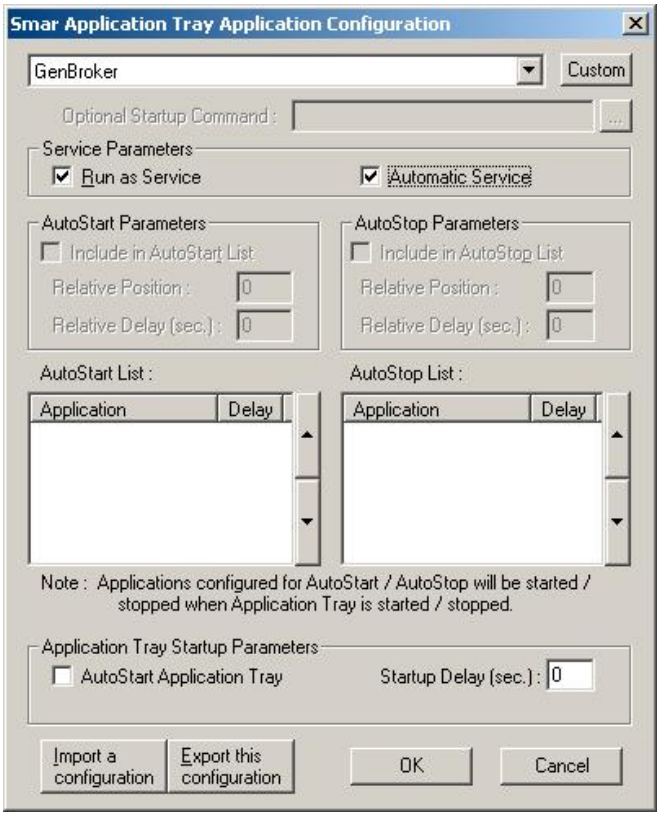
2. Once an application is converted to an NT service, it can optionally be configured to be an **Automatic** service so that it will automatically start every time the PC is rebooted, as shown in the figure below.



Registering an Automatic NT Service

You can also configure the NT service parameters by selecting **Configure** on the ProcessView Tray user interface. This opens the **ProcessView Tray Application Configuration** dialog box, as shown in the figure below. For those applications that can be run as an NT service, select the application from the drop-down list at the top-left corner of the dialog box and check the **Run As Service** check box and the optional **Automatic Service** check box.

NOTE
If the selected application cannot run as an NT service, the NT service options will be grayed out. When an application is run as an automatic NT service, the auto start and auto stop options for that application are disabled.



Configuring NT Service Parameters

OLE Automation

OLE Automation support was originally integrated into ProcessView Tray in Version 6.0. The **IscriptCommands** interface allows Visual Basic and other clients to start and stop individual ProcessView applications as well as make changes to the existing ProcessView Tray configuration.

Methods
GetRegisteredApps(AppRefIDs, AppDescriptions)
Inputs
none

Outputs
AppRefIDs is a string array of reference IDs that are used to access individual components.
AppDescriptions is a string array of descriptions that correspond to the AppRefIDs.
Description
Obtains a list of registered ProcessView Tray applications. The AppRefIDs are the identifiers to be used to retrieve other information as described below.

Startup()
Inputs
none

Outputs
none

Description
Starts all components currently configured to AutoStart.

Shutdown()

Inputs
none

Outputs
none

Description
Shuts down all components currently configured to AutoStop.

Start(AppRefID)
Inputs
AppRefID: String specifying application identifier obtained using GetRegisteredApps().

Outputs
none

Description
Starts the specified component.

Stop(AppRefID)
Inputs
AppRefID: String specifying application identifier obtained using GetRegisteredApps().

Outputs
none

Description
Shuts down the specified component.

NOTE: A deadlock will occur if a script is used to run the ProcessView Tray **Stop(AppRefID as String)** method because the script will attempt to stop itself (e.g. if GraphWorX is used to run a script that instructs ProcessView Tray to stop GraphWorX).

GetStartTableEntry(AppRefID, Position, RelativeStartDelay)
Inputs
AppRefID: String specifying application identifier obtained using GetRegisteredApps().

Outputs
Position: Long value that specifies the position in the startup table.
RelativeStartDelay: Long value specifying time in seconds to wait before starting application.

Description
Gets the current start table entries as configured in ProcessView Tray.
SetStartTableEntry(AppRefID, Position, RelativeStartDelay)
Inputs
AppRefID: String specifying application identifier obtained using GetRegisteredApps().
Position: Long value that specifies the position in the startup table.
RelativeStartDelay: Long value specifying time in seconds to wait before starting application.

Description
Sets the current start table entries in ProcessView Tray.
GetStopTableEntry (AppRefID, Position, RelativeStoptDelay)
Inputs
AppRefID: String specifying application identifier obtained using GetRegisteredApps().

Outputs
Position: Long value that specifies the position in the startup table.
RelativeStartDelay: Long value specifying time in seconds to wait before starting application.

Description
Gets the current stop table entries as configured in ProcessView Tray.

SetStopTableEntry (AppRefID, Position, RelativeStoptDelay)
Inputs
AppRefID: String specifying application identifier obtained using GetRegisteredApps().
Position: Long value that specifies the position in the startup table.
RelativeStartDelay: Long value specifying time in seconds to wait before starting application.

Description
Sets the current stop table entries in ProcessView Tray.

Properties

AutoStartState(AppRefID)
Inputs
AppRefID: String specifying application identifier obtained using GetRegisteredApps().

Description
Boolean value reflects the specified application's AutoStart state.

AutoStopState(AppRefID)
Inputs
AppRefID: String specifying application identifier obtained using GetRegisteredApps().

Description
Boolean value reflects the specified application's AutoStop state.

ServiceState(AppRefID)
Inputs
AppRefID: String specifying application identifier obtained using GetRegisteredApps().

Description
Boolean value reflects the specified application's Service state.

AutomaticState(AppRefID)
Inputs
AppRefID: String specifying application identifier obtained using GetRegisteredApps().

Description
Boolean value reflects the specified application's Automatic state when configured as a Service.

RunsAsService(AppRefID)
Inputs
AppRefID: String specifying application identifier obtained using GetRegisteredApps().

Description
Boolean value reflects whether the specified application can run as a service.

IsRunning(AppRefID)
Inputs
AppRefID: String specifying application identifier obtained using GetRegisteredApps().

Description
Boolean value reflects the specified application's current runtime status.

SupportsCommand(AppRefID)
Inputs
AppRefID: String specifying application identifier obtained using GetRegisteredApps().

Description
Boolean value reflects whether the specified application supports a command line string.

StartCommand (AppRefID)
Inputs
AppRefID: String specifying application identifier obtained using GetRegisteredApps().

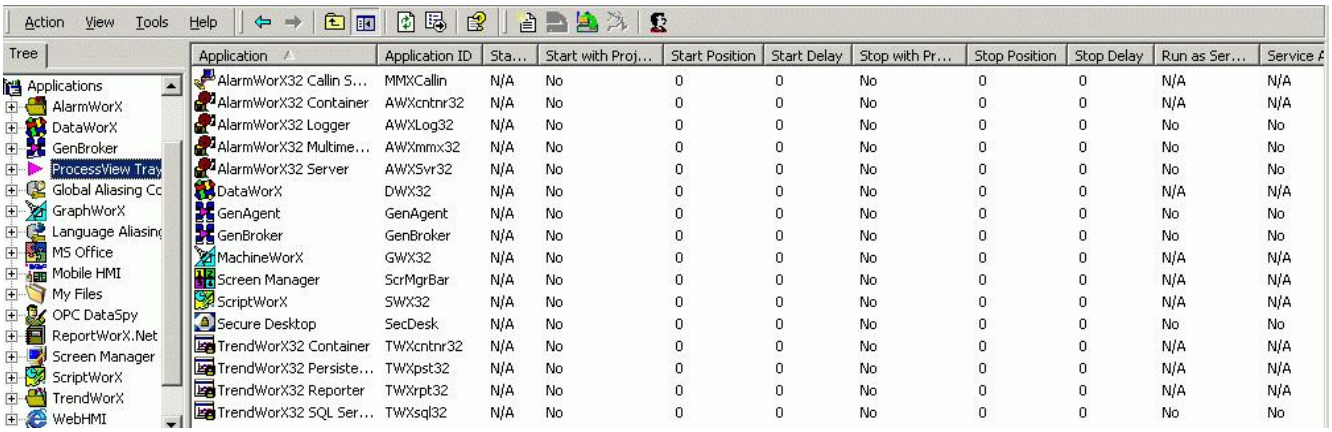
Description
String value reflects the specified application's command line string.

Using ProcessView Tray in ProjectWorX

The **ProcessView Tray** tree of the ProjectWorX console, shown in the figure below, includes all the capabilities of the ProcessView ProcessView Tray application. When the ProcessView Tray tree is selected, a list of the ProcessView applications on the machine is provided, as shown in the figure below. From this list you can check to see whether a server or an application is running and browse their settings.

ProcessView Tray provides a way to either manually or automatically start and/or stop applications within ProcessView. In the ProcessView Tray tree for the active project in the ProjectWorX console, you can specify which applications and servers to start and stop. You can also set up applications to run auto start upon Windows startup, setting their timing and launching position parameters.

NOTE
An application can be detected as running only if the application is launched through the ProcessView Tray. The ProcessView Tray tree shows data for the active project only.

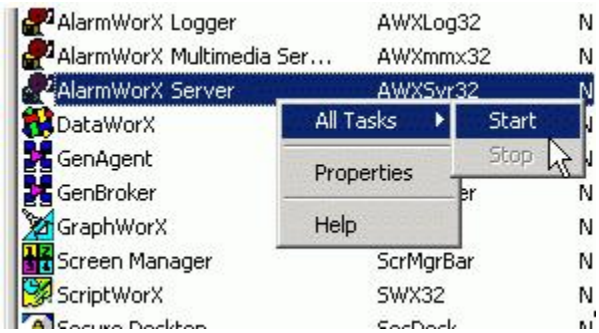


ProcessView Tray Tree in ProjectWorX Console

Starting Applications in ProjectWorX

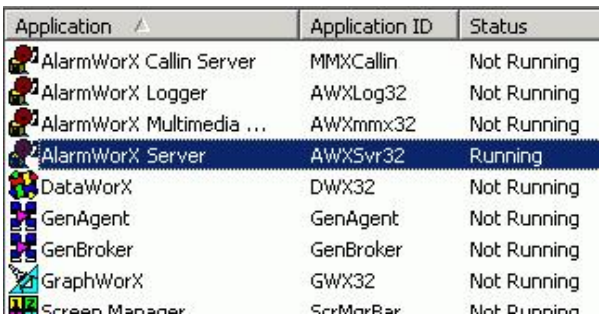
To launch an application from ProcessView Tray:

1. Right-click the application in the details view and select **All Tasks > Start** from the pop-up menu, as shown in the figure below.



Starting an Application in ProjectWorX

2. The application is launched and the **Status** changes to **Running,** as shown in the figure below.

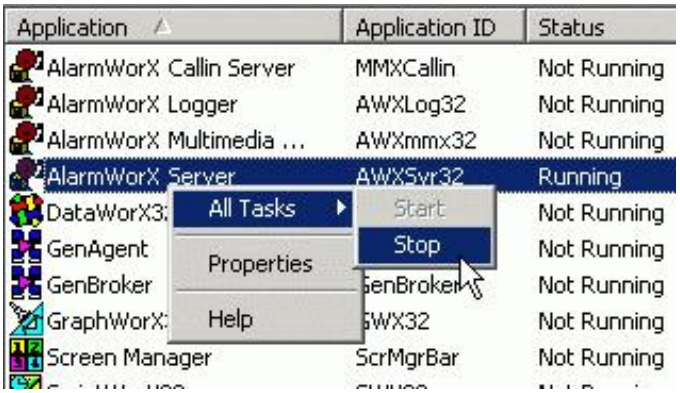


Application Running

Stopping Applications in ProjectWorX

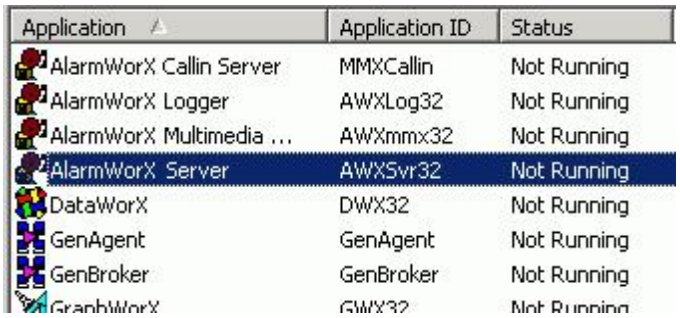
To shut down a running application from ProcessView Tray:

1. Right-click the application in the details view and select **All Tasks > Stop** from the pop-up menu, as shown in the figure below.



Stopping an Application in ProcessView Tray

2. The application is launched, and the status changes to **Not Running,** as shown in the figure below.



Application Stopped in ProcessView Tray

Setting Application Properties in ProjectWorX

From the ProcessView Tray tree of the ProjectWorX console, you can view application properties for both clients and servers. Any changes that you make to the application properties settings are reflected in the details list of the ProjectWorX ProcessView Tray tree. To change the properties for an application, right-click the application in the details view and select **Properties** from the pop-up menu, as shown in the figure below.

Viewing Application Properties in ProjectWorX

The **Properties** dialog box opens, as shown in the figure below. The properties indicate the current status of the application (i.e. Running or Not Running).

The server properties are slightly different from the client properties. The server properties have **Run as a Service** and **Run at Startup** options, while client properties instead have a **Command Line** feature that indicates the designated startup file for the application (as specified in the active project).

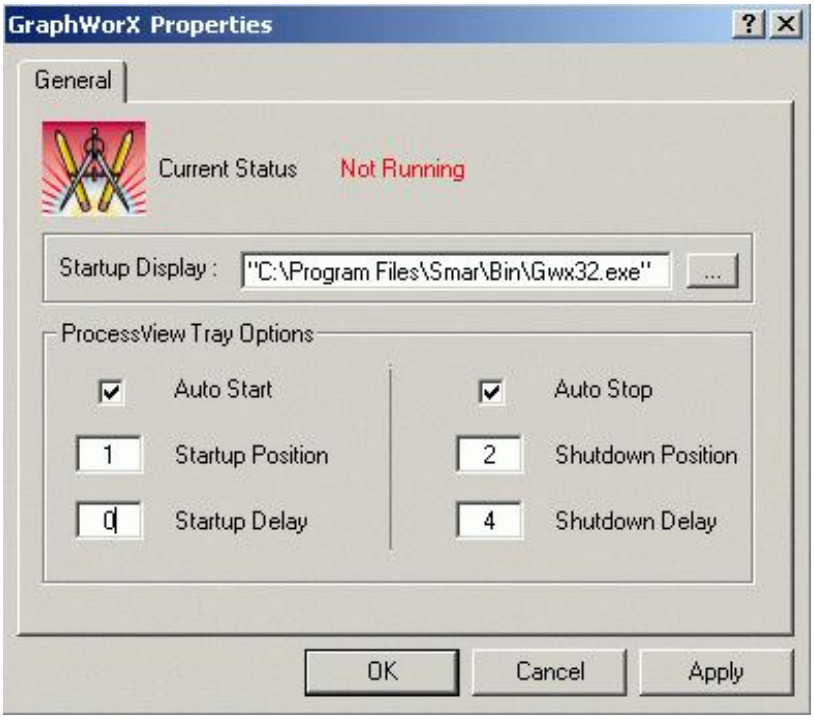
ProcessView Tray enables you register and run servers as an NT service. To convert a server to an NT service, check the **Run as a Service** check box, as shown in the figure below. Once an application is converted to an NT service, checking the **Run at Startup** check box makes it an automatic service so that it will automatically start every time the computer is restarted.

Server Properties

The **Auto Start** and **Auto Stop** options enable a ProcessView system application to be opened and shut down automatically and in an orderly fashion. Selecting these items puts the application in a management list with other applications that are auto started and auto stopped. The order in which these applications are started or stopped is determined by the relative start and stop positions specified in the application properties. Both the server properties and client properties dialogs have the following ProcessView Tray application startup and shutdown options, which set the start and stop parameters for the application:

- **Auto Start:** When checked, ProcessView Tray automatically launches the application upon system startup.
- **Startup Position:** Specifies the order in which the application will be started (if multiple applications are auto started). For example, if both GraphWorX and AlarmWorX are to be auto started but you want GraphWorX to start before AlarmWorX starts, set the startup position for GraphWorX to "1" and the startup position for AlarmWorX to "2."
- **Startup Delay:** Sets the delay (in seconds) before the application is launched.
- **Auto Stop:** When checked, ProcessView Tray automatically stops the application upon system shutdown.
- **Shutdown Position:** Specifies the order in which the application will be shut down (if multiple applications are auto stopped).
- **Shutdown Delay:** Sets the delay (in seconds) before the application is stopped.

Click the **Apply** button to set the parameters you have selected in the application properties.



Client Properties

Viewing Application Parameters in ProjectWorX

The parameters that are set the application properties are displayed in the details view of the ProcessView Tray, as shown in the figure below. The column headers indicate the following properties:

- **Application:** Name of the application.
- **Application ID:** Name of the application's executable file.
- **Status:** Current application status (i.e. Running or Not Running).
- **Start With Project:** States whether auto start is enabled for the application (i.e. Yes or No).
- **Start Position:** Specifies the order in which the application will be started (if multiple applications are auto started).
- **Start Delay:** Delay (in seconds) before the application is launched.
- **Stop With Project:** States whether auto stop is enabled for the application (i.e. Yes or No).
- **Stop Position:** Specifies the order in which the application will be shut down (if multiple applications are auto stopped).
- **Stop Delay:** Delay (in seconds) before the application is stopped.
- **Run as Service:** States whether a server is run as an NT service (i.e. Yes or No).
- **Service Autorun:** States whether a server run as an NT service will automatically start every time the computer is restarted (i.e. Yes or No).
- **Command Line:** Indicates the designated startup file/database for the application (as specified in the active project).

Start with Project	Start Position	Start Delay	Stop with Project	Stop Position	Stop Delay	Run as Service	Service Autorun
No	0	0	No	0	0	No	No
No	0	0	No	0	0	No	No
No	0	0	No	0	0	No	No
No	2	4	No	1	0	Yes	Yes
No	0	0	No	0	0	N/A	N/A
No	0	0	No	0	0	No	No
No	0	0	No	0	0	No	No
Yes	1	0	Yes	2	4	N/A	N/A
No	0	0	No	0	0	N/A	N/A
No	0	0	No	0	0	N/A	N/A
No	0	0	No	0	0	No	No
No	0	0	No	0	0	N/A	N/A
No	0	0	No	0	0	N/A	N/A
No	0	0	No	0	0	No	No

Application Parameters in ProjectWorX